```
#########################################################################
#
# Generation of Class 1 by Fractions
#
#########################################################################
#
main.1.F <- function(system, maxn){
#
f <- 1
for(n in 1:maxn){
#
  if(system == 1){
#
# Calkin-Wilf
#
    f[2*n  ] <-  f[n]    / (f[n]+1)
    f[2*n+1] <- (f[n]+1) /       1
  }
#
  if(system == 2){
#
# driB
#
    f[2*n  ] <-       1  / (f[n]+1)
    f[2*n+1] <- (f[n]+1) /  f[n]
  }
#
  if(system == 3){
#
# Yu-Ting
#
    f[2*n  ] <-  f[n]    / (f[n]+1)
    f[2*n+1] <- (f[n]+1) /  f[n]
  }
#
  if(system == 4){
#
# Yurramendi-1
#
    f[2*n  ] <-       1  / (f[n]+1)
    f[2*n+1] <- (f[n]+1) /  f[n]
  }
#
}
#
  return(f)
#
}
#
#########################################################################
#
# Examples: Build the beginning of all four systems
#
library(MASS)
#
fractions(main.1.F(system = 1, maxn = 31)) # CW
fractions(main.1.F(system = 2, maxn = 31)) # driB
fractions(main.1.F(system = 3, maxn = 31)) # Yu-Ting
fractions(main.1.F(system = 4, maxn = 31)) # Yurramendi-1
#
#########################################################################
```